The Rise of Threat Analysis and the Fall of Compliance in Mitigating Web Application Security Risks

Tony Ucedavelez
OWASP Atlanta Chapter Lead
tonyuv@versprite.com

OWASP
LA ISACA Meeting
Feb 2010 Meeting

Copyright 2009 © The OWASP Foundation
Permission is granted to copy, distribute and/or modify this document under the terms of the OWASP License.

The OWASP Foundation
http://www.owasp.org

Meeting Agenda

- Status quo of regulatory compliance in mitigating risk
- Threat modeling techniques for cybercrime threats
- Attack tree analysis
- Attack vectors analysis
- Architecture threat and countermeasures analysis via threat modeling
- Mitigation strategies against cybercrime attacks

Status Quo of Security Policy and Regulatory Compliance in Mitigating Risks

Heartland-Hannaford-TJ Maxx Identity Theft and Fraud Cases

- **Albert Gonzales Federal Indictment facts**:
  - **Theft of 130 Million Credit Card Accounts** (largest ever reported)
  - **Exploited SQL Injection vulnerabilities** to install malware
  - **Perpetrated by a Cybercrime gang** (Darkmarket) profited from the sale of ACC#, PINs to fake credit cards and commit ATM fraud (Darkmarket)
- **Security compliance facts**:
  - Hannaford was PCI compliant
  - TJX Maxx was fined for not being PCI compliant during the breach

Let's Quantify The Losses: TJ Maxx

- The cyberattack on the retailer Marshalls and TJ Maxx (part of the TJX Companies) that was disclosed in January 2007 **resulted in the company recording an after-tax cash charge of approximately $118 million**, or $.25 per share.

- **The company increased its estimate of pre-tax charges for the compromise of nearly $216 million** from an earlier projection of approximately $168 million.

- According to some experts, **TJX may have to spend in the end a total of more than $500 million**, including litigation fees and government fines.

Can We Estimate the Potential Business Impact of SQL Injection Attack?

- Simple risk correlation facts
  - 691 $ per identity theft incident (Javelin)
  - 13% of incidents involve breach of web channel (DataLossDB.org)
  - 19% of all attacks use SQL injection as attack vector (WHID)

- The probability that a SQL injection vulnerability will cause sensitive data loss is approx 2.5 %

- The potential impact of SQL injection resulting in loss of 14 million customer's PII served through the web channel:
  - **0.025 X 14,000,000 X 691 = $ 241 ML**

How Compliance Drives Security?

- Regulations such as PCI, Gramm-Leach Bliley Act (GLBA), FFIEC, HIPAA, SB 1386, AB 1950 drive security via an adversarial approach:
  - **Fail audit by one of the regulatory compliance bodies results in additional fines, restrictions and controls**
  - **Compliance regs are typically vague and general, leaving too much room for interpretation for them to be business topical**
  - **Leak of PII prompts public information disclosure in most US states (SB1386)**
    - **Fear of backlash, private suits, etc**

How Compliance Drives Security? (Cont)

- PCI DSS brings precision, but with the traditional FUD
  - **Greater detail surrounding technical / non-tech security requirements**
  - **Standard compliance FUD is main security driver:**
    - **Running afoul of PCI means you can't do business using credit cards; losing accredidation equates to opportunity costs** (ex: can't do business with Wal-Mart)
    - **Hefty fines**
    - **Continues to evolve to encompass greater forethought as to what should be protected for cardholder data.**

PCI DSS And Protection of Sensitive Data

Technical Guidelines for PCI Data Storage

	Data Element	Storage Permitted	Protection Required	PCI DSS Req. 3.4
Cardholder Data	Primary Account Number (PAN)	Yes	Yes	Yes
	Cardholder Name[1]	Yes	Yes[1]	No
	Service Code[1]	Yes	Yes[1]	No
	Expiration Date[1]	Yes	Yes[1]	No
Sensitive Authentication Data[2]	Full Magnetic Stripe Data[3]	No	N/A	N/A
	CAV2/CVC2/CVV2/CID	No	N/A	N/A
	PIN/PIN Block	No	N/A	N/A

[PCI-DSS] 3.3 Mask PAN when displayed (the first six and last four digits are the maximum number of digits to be displayed).

[PCI-DSS] 3.4 Render PAN, at minimum, unreadable anywhere it is stored (including on portable digital media, backup media, in logs)

REALITY
CHECK
AHEAD

Underground economy for stolen credit card and bank account credentials

2008 Rank	2007 Rank	Item	2008 Percentage	2007 Percentage	Range of Prices
1	1	Credit card information	32%	21%	$0.06–$30
2	2	Bank account credentials	19%	17%	$10–$1000
3	9	Email accounts	5%	4%	$0.10–$100
4	3	Email addresses	5%	6%	$0.33/MB–$100/MB
5	12	Proxies	4%	3%	$0.16–$20
6	4	Full identities	4%	6%	$0.70–$60
7	6	Mailers	3%	5%	$2–$40
8	5	Cash out services	3%	5%	8%–50% or flat rate of $200–$2000 per item
9	17	Shell scripts	3%	2%	$2–$20
10	8	Scams	3%	5%	$3–$40/week for hosting, $2–$20 design

Table 1. Goods and services available for sale on underground economy servers
Source: Symantec

A Critical View of Compliance

- Does compliance makes us more secure?
  - Plenty of PCI compliant firms have recently been hit with major security breaches
  - Blindly checking off compliance requirements is neither cost-effective nor risk effective
  - Too much confidence in compliance regulations & even frameworks creates a schism apart from security strategy
    - Makes security practical
    - Boxed thinking; no ingenuity
    - C-Levels question value to what they perceive as 'extra' or 'misguided' efforts

A Critical View of Info_Sec

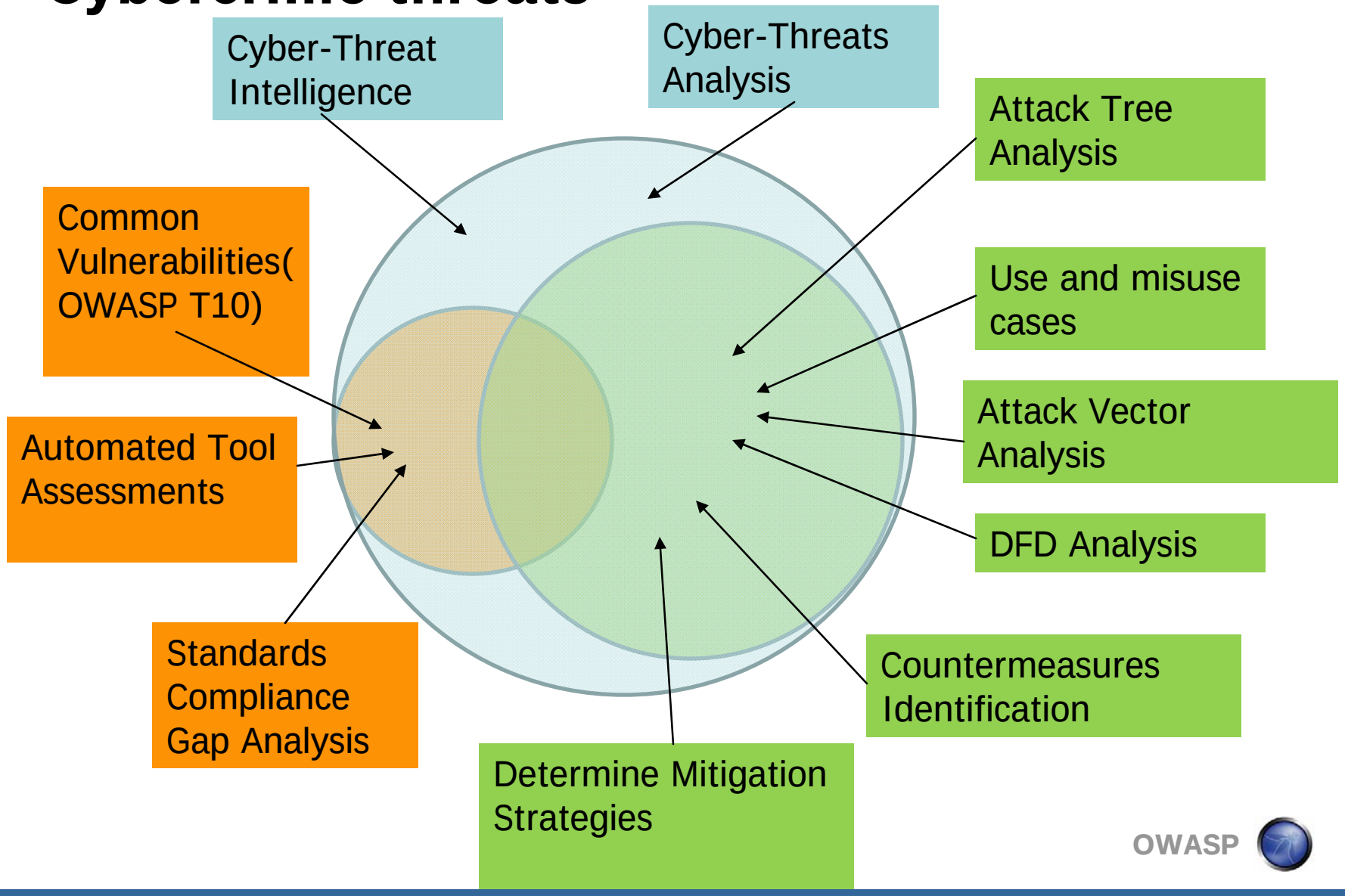
- Follow the leader syndrome
  - Too much 'What is BofA doing?' or 'What does Gartner say?'
  - Who cares. Build you(Devil's Advocate) – This is sage advice but in moderation – r own security strategy per your own unique business objectives, security architecture, and invested technologies.
- Temper sage advice
  - "The conclusion is unavoidable: any notion that security is a matter of simply protecting the network perimeter is hopelessly out of date" - *IDC and Symantec, 2004*
  - Develop parallel strategies at the application layer while not neglecting the network!

Non Compliance From Risk Perspective

Regulatory noncompliance is by it self a business risk, so a disregard to compliance is not the intended message: assessing the likelihood and potential costs of a particular threat against the cost of preventing or mitigating that threat

Threat modeling techniques for cybercrime threats

Application Threat Modeling (ATM) For Cybercrime threats



Threat Analysis via ATM

- Application risk model to investigate application threats, determine risk and make informed decisions prior to production
- Attack and Threat Analysis
  - Attack Trees
  - Use and Misuse Cases
  - Data flow diagrams
- Technical Risk Analysis
  - Validate attack scenarios
  - Identify vulnerabilities/ countermeasures beforehand
  - Mitigate risk based upon business impact

Cyber-threat Intelligence via ATM

- Goals
  - **Understand cyber threats and how they might affect your business**
    - Correlate cyber threats to industry segment and business type
    - Threat management gives way to attack libraries
  - **Learn from cyber criminals motives and likely attack patterns**:
    - Leverage attack trees to learn probable attack vectors to be used for perceived threat scenarios
  - **Plan defenses for the attack vectors being used**:
    - Identify the efficiency in which countermeasures can be devised to thwart attack scenarios
    - Determine gaps in present countermeasures in mitigating application risks

Reported: Cyber attacks exploiting insecure MS SQL Configuration

1. Use "xp_cmdshell“ to download hacker tools to the compromised MSSQL server.
2. Obtain valid Windows credentials by using fgdump
3. Install network "sniffers" to identify card data and systems involved in processing credit card transactions.
4. Install backdoors that "beacon" periodically to their command and control servers
5. Target databases, Hardware Security Modules (HSMs), and processing applications in an effort to obtain credit card data or brute-force ATM PINs.
6. Use WinRAR to compress the information they pilfer from the compromised networks.

Source: IC3 – Internet Crime Complaint Center

Reported: Drive-By Download Attacks

1. **Compromise of a legitimate web site**
   - Do Google search for asp + “parameter=“ and send SQL Injection Malware to all targets (hosts), the malware is an hidden IFRAME
   - When finds a target (host) vulnerable to SQL injection installs the malware

2. **User visits the legitimate site**
   - The hidden IFRAME redirects the browser to malicious site
   - Malicious content is downloaded exploiting browser plug-in vulnerabilities such as ADOBE FLASH

3. **Malicious code is installed on the user’s computer**
   - When downloadable browser content is executed (e.g. multimedia player) the PC is infected with malware/spyware

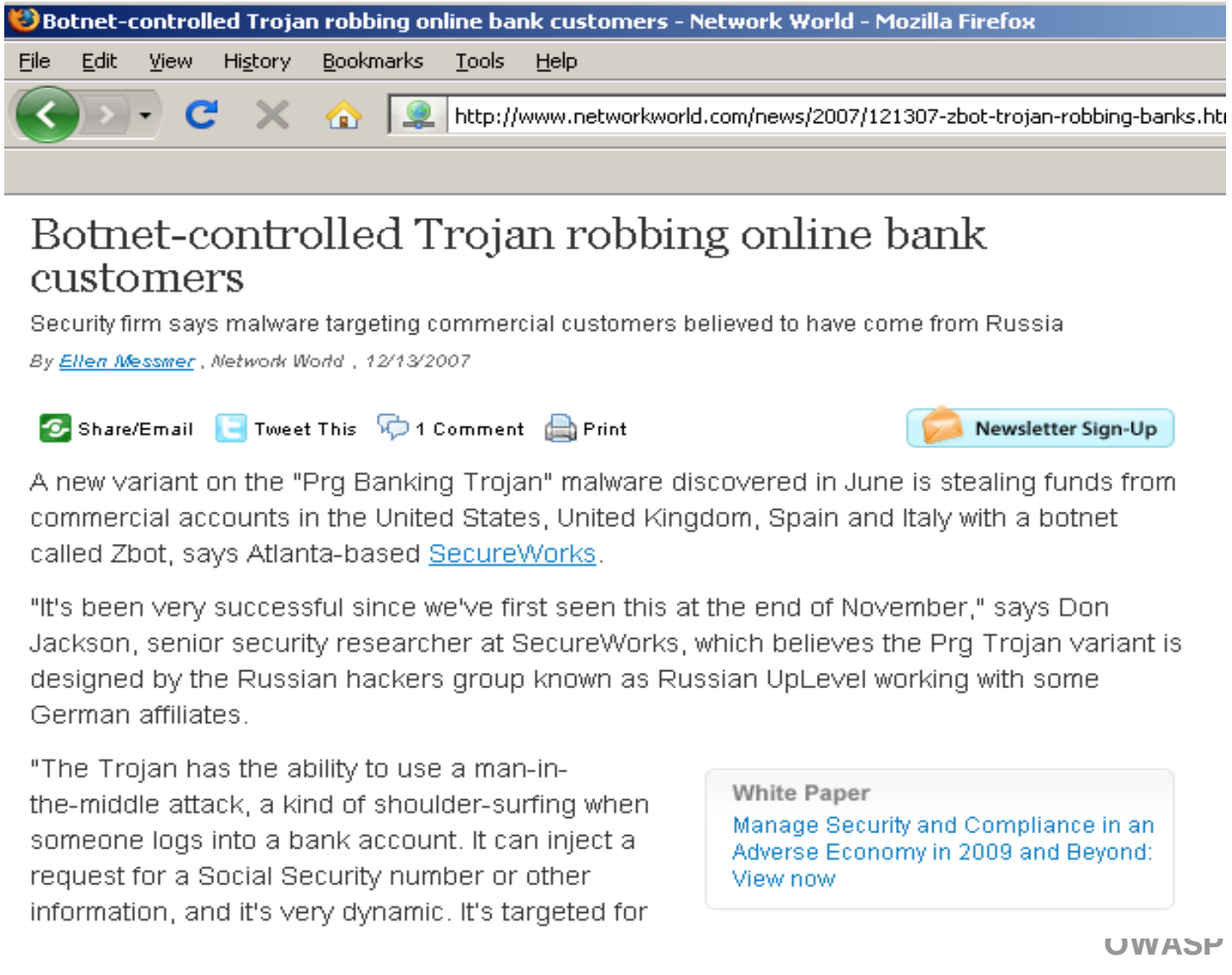
Cybercrime's Leverage of Browser Based Vulnerabilities is related to favorable odds

- ***"Profit motivated cyber-criminals have rapidly adopted web browser exploitation as a key vector for malware installation"*** - *ETH Zurich Tech Report Nr. 288*

- **19 % of web published/reported vulnerabilities are web browser vulnerabilities** – *Cenzic Web Application Security Quarterly Report Q3-Q4 2008*
  - 2 % are second order web server vulnerabilities
  - 19 % are browser vulnerabilities (include plugins & ActiveX)
  - 79 % are web application vulnerabilities

- Cybercriminals research and test most common application exploits for greatest ease in compromising sensitive financial and personal information

Cybercrime Attacks Against Online Bank Customers


Botnet-controlled Trojan robbing online bank customers - Network World - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://www.networkworld.com/news/2007/121307-zbot-trojan-robbing-banks.ht
Botnet-controlled Trojan robbing online bank customers
Security firm says malware targeting commercial customers believed to have come from Russia
By Ellen Messmer , Network World , 12/13/2007
Share/Email
Tweet This
1 Comment
Print
Newsletter Sign-Up
A new variant on the "Prg Banking Trojan" malware discovered in June is stealing funds from commercial accounts in the United States, United Kingdom, Spain and Italy with a botnet called Zbot, says Atlanta-based SecureWorks.
"It's been very successful since we've first seen this at the end of November," says Don Jackson, senior security researcher at SecureWorks, which believes the Prg Trojan variant is designed by the Russian hackers group known as Russian UpLevel working with some German affiliates.
"The Trojan has the ability to use a man-in-the-middle attack, a kind of shoulder-surfing when someone logs into a bank account. It can inject a request for a Social Security number or other information, and it's very dynamic. It's targeted for
White Paper
Manage Security and Compliance in an Adverse Economy in 2009 and Beyond: View now

Cybercrime Attacks Again ATMs

Cybercriminals Refine Data-sniffing Software for ATM Fraud

Jeremy Kirk, IDG News Service

Thursday, June 04, 2009 3:40 AM PDT

Cybercriminals are improving a malicious software program that can be installed on ATMs running Microsoft's Windows XP operating system that records sensitive card details, according to security vendor Trustwave.

The malware has been found on ATMs in Eastern European countries, according to a Trustwave report.

The malware records the magnetic stripe information on the back of a card as well as the PIN (Personal Identification Number), which would potentially allow criminals to clone the card in order to withdraw cash.

The collected card data, which is encrypted using the DES (Data Encryption Standard) algorithm, can be printed out by the ATM's receipt printer, Trustwave wrote.

The malware is controlled via a GUI that is displayed when a so-called "trigger card" is inserted into the machine by a criminal. The trigger card causes a small window to appear that gives its controller 10 seconds to pick one of 10 command options using the ATM's keypad.

"The malware contains advanced management functionality allowing the attacker to fully control the compromised ATM through a customized user interface built into the malware," Trustwave wrote.

PEOPLE WHO READ THIS ALSO READ:

Judge Dismisses NSA Wiretap Civil Liberties Suits

Mozilla Delays Firefox 3.5 Release Candidate

Judge Sides With UK Bank in 'phantom Withdrawal' Case

Flying Car Passes Its Test Flight

HP Wants You to Turn off Your PC

Ming Dynasty Vase With Intel Atom, Anyone?

Recommendations by loomia

FIND A REVIEW

Cybercrime Counter-Intelligence Goals

- **Understand cyber threats and how they may affect your business**:
  - What cyberthreats are relevant to your industry?
    - Peel the threat onion (industry, geographic, local market, overall business, branch)

  Are you looking at outdated cyber threats?
- **Learn from cyber criminals motives and the most likely attack scenarios**:
  - Become your enemy. Build the right attack tree to walk through probable attack scenarios.
  - What are their business drivers for an attack on your infrastructure?

    Religious, political, financial, personal, other beliefs, etc
- **Plan defenses for the attack vectors being used**:
  - Based on the likely attack patterns for each branch of the attack tree, identify which application vulnerabilities can be exploited via which exploits

Attack Tree Analysis

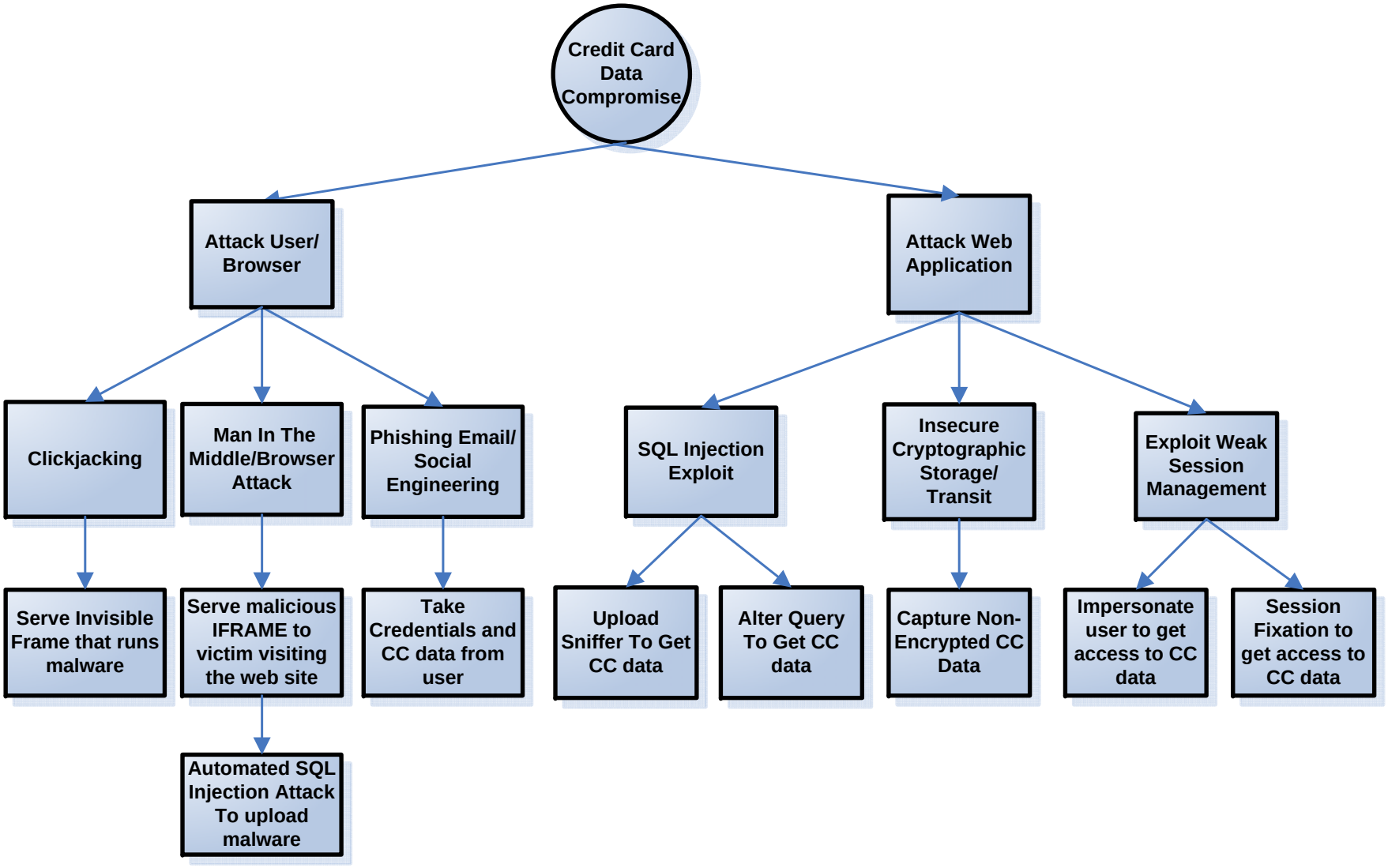
Attacker Motives and Goals

- Define threat agent attributes
  - Hostile (fraudster, theft) vs. non hostile (untrained employee)
  - External (user/customer, visitor) vs. internal (intranet user/employee
- Define common motives
  - Fraudster: financial and monetary gain
  - Thief: theft of IP, PII or business data
- Outcomes
  - Theft
  - Business advantage
  - Damage (tangible and intangible)

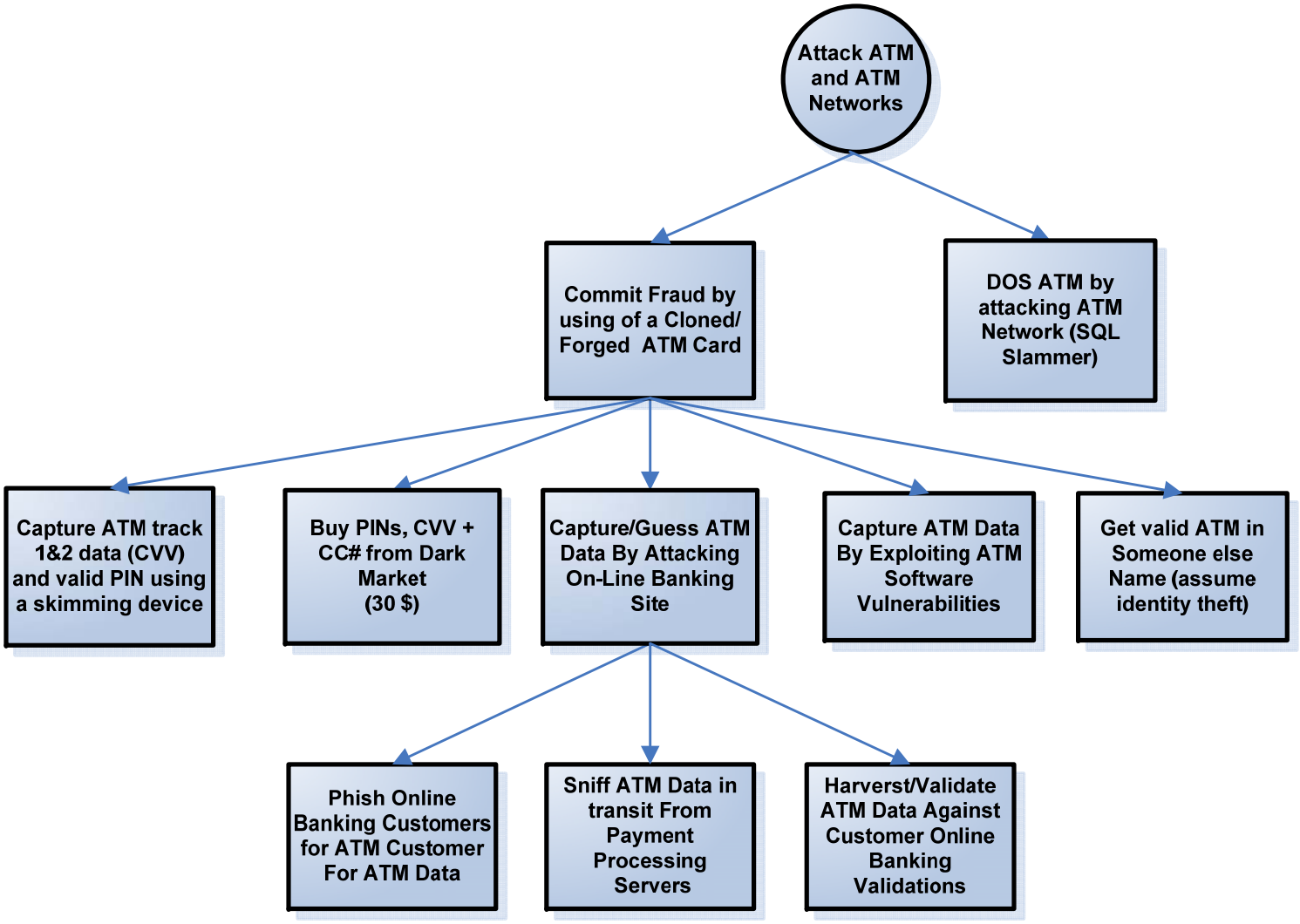
Attack Trees

- Formal methodology for analyzing the security of systems and subsystems
- They provide a way to think about security from the attacker perspective
- The root node represents the target of the attack, while the leaf nodes represent the means for reaching the target, which are the events that comprise the attack

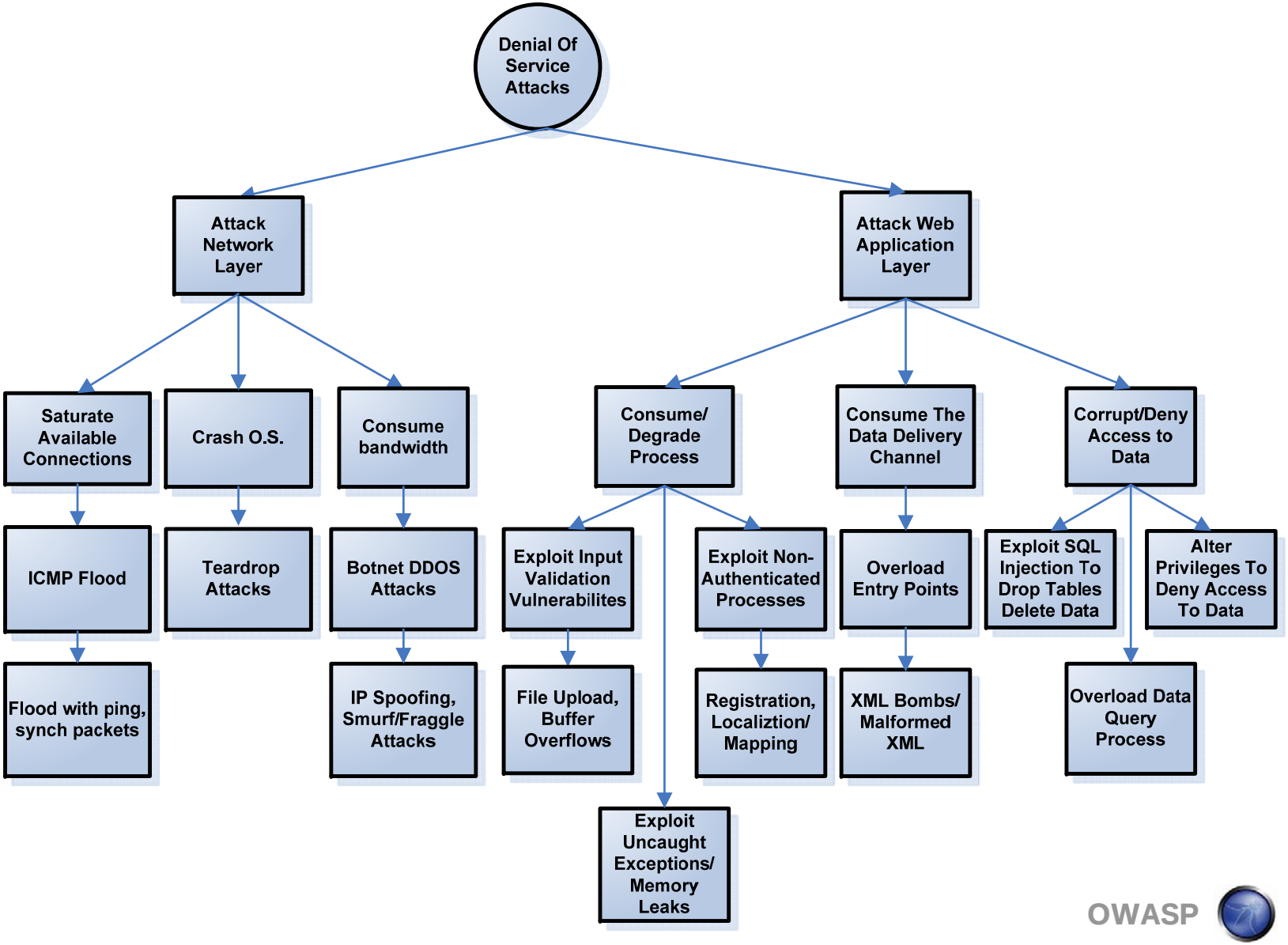
Threat Tree For Credit Card Attacks


Credit Card Data Compromise
Attack User/ Browser
Attack Web Application
Clickjacking
Man In The Middle/Browser Attack
Phishing Email/ Social Engineering
SQL Injection Exploit
Insecure Cryptographic Storage/ Transit
Exploit Weak Session Management
Serve Invisible Frame that runs malware
Serve malicious IFRAME to victim visiting the web site
Take Credentials and CC data from user
Upload Sniffer To Get CC data
Alter Query To Get CC data
Capture Non-Encrypted CC Data
Impersonate user to get access to CC data
Session Fixation to get access to CC data
Automated SQL Injection Attack To upload malware

Threat Tree For ATM Attacks


Attack ATM and ATM Networks
Commit Fraud by using of a Cloned/ Forged ATM Card
DOS ATM by attacking ATM Network (SQL Slammer)
Capture ATM track 1&2 data (CVV) and valid PIN using a skimming device
Buy PINs, CVV + CC# from Dark Market (30 $)
Capture/Guess ATM Data By Attacking On-Line Banking Site
Capture ATM Data By Exploiting ATM Software Vulnerabilities
Get valid ATM in Someone else Name (assume identity theft)
Phish Online Banking Customers for ATM Customer For ATM Data
Sniff ATM Data in transit From Payment Processing Servers
Harverst/Validate ATM Data Against Customer Online Banking Validations

Threat Tree For Denial Of Service Attacks



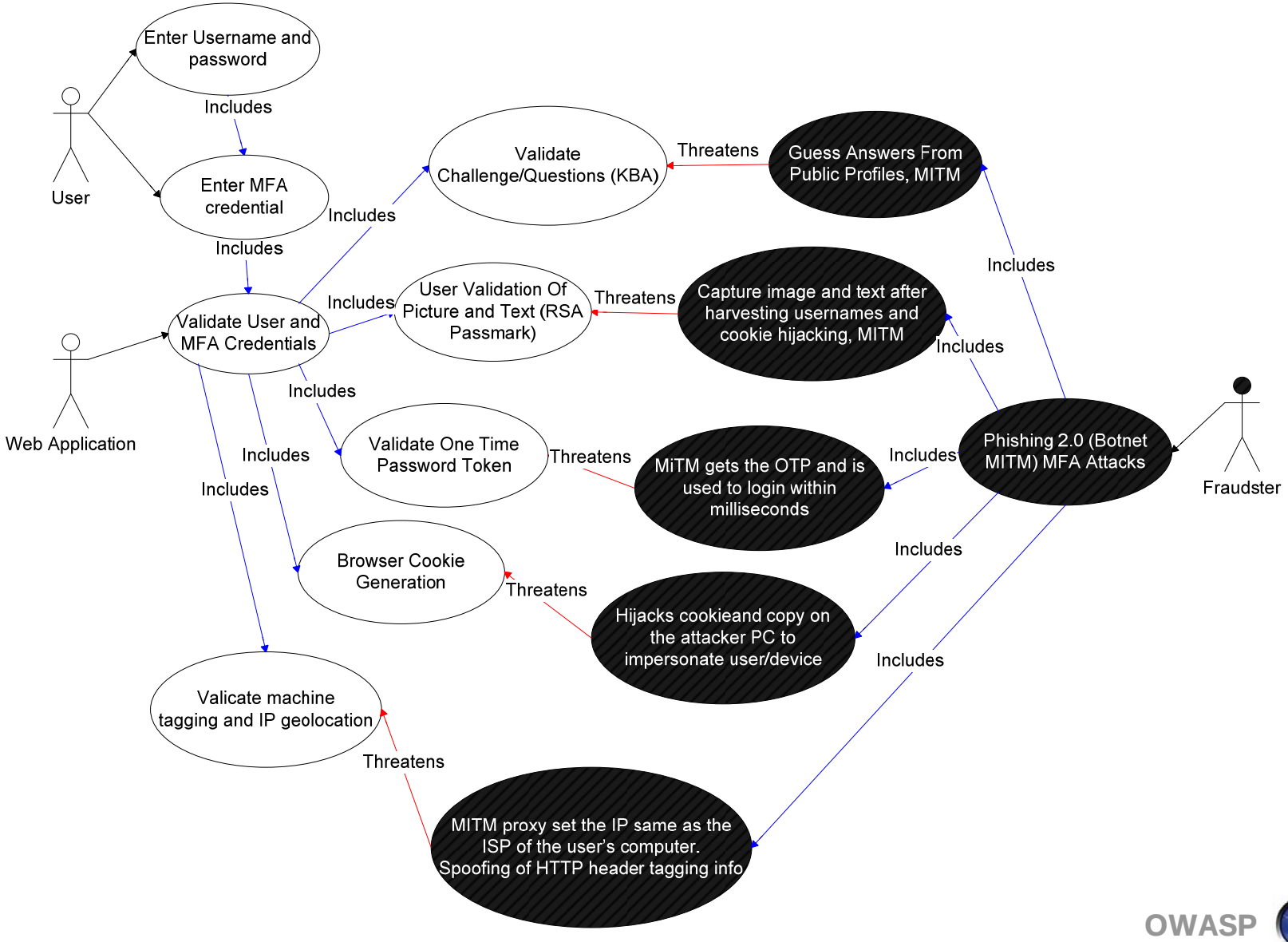
Use and Abuse Cases

Use and Abuse Cases

- Use cases describe the use of application and resources as intended functionality through a sequence of steps/actions
- A misuse case describes the use of application and resources for any other purpose than for that which they were originally intended
- Use and abuse cases are useful for
  - Derive requirements for security controls
  - Select countermeasures to mitigate attacks
  - Test cases for security testing

Use and Abuse Of Logins

Use And Abuse Of MFA



Use and Misuse Cases For Authorization Bypass

- AUTHORIZATION 1.0
- **Use Case Name:** User authenticates to the application by supplying username and password through the log-on web page
- **Misuse Case Name**: Rogue user changes role ID parameter to elevate his privileges to administrator
- **Threat Event:** Rogue employee seeks unauthorized administrator privileges

Use and Misuse Cases For Authorization Bypass: Text Form

- **Use Case:**
  - User with user role authenticates with the application
  - RoleID is sent from the server to the client upon authentication in the hidden field
  - User is granted view permissions by the server based upon the Role ID
- **Misuse Case:**
  - Rogue employee authenticates with user credentials
  - RoleID is sent from the server to the client upon authentication of the user in the hidden field
  - Rogue employee capture Role ID back to the server with a web proxy and modifies it to admin value (e.g. Role ID=5) before being processed by the server
  - Rogue employee is granted admin permissions by the server

Attack Vector Analysis

Understanding Attack Vectors

- Don't confuse attack vectors with the payload that is carried out
  - **Attack vectors**: channel in which attack is conducted over
    - email, attachments, worms, web pages, downloads, deception (aka social engineering), hackers
  - **Payloads**: malicious binary data in viruses, spyware, trojans, malicious scripting/executables.
- XSS Example:
  - The attack vector with a payload consisting in a script (**also encoded**) to capture sensitive information (e.g. cookie stored on the browser) such as in an alert dialog:
  - *http://server/cgibin/testcgi.exe?<SCRIPT>alert("Cookie"+document.cookie)</SCRIPT>*

Attack Vectors Survey

- Remote access services and web applications were the attack targets through which the attacker gained access to corporate systems in the vast majority of cases.(Verizon 2009 data breach report)
- Identify common attack vectors with web applications:
  - Non Privileged web service ports for CMS – WordPress, Joomla, Mambo vulnerabilities
  - Traditional HTTP Requests
    - PHP Remote File Include (*allow url fopen*)
    - SQL Injection
    - XSS/ XSRF
  - SSL Based Attacks
    - MITM
    - PKCS Padding Attacks
  - Multiple or hybrid attack vectors
    - Phishing, Vishing

Understanding the Attacker via DFDs

DFDs create an application walk through of possible attack scenarios

- Identify ***first order injection*** and ***second-order injection*** (e.g. to attack resources directly)
- Isolate attacks by groups:
  - Platforms, Software Arch (J2EE, .NET)
  - Communication channels (SSL, IPSec, IRC, RPC)
  - Type of Attack (Spoofing, DoS, Tampering of Data, Elevation of Privileges, etc)
- Correlate attacks to vulnerabilities
  - Branches within the DFD will depict vulns needed to realize the threat
- Correlate attack to actors (human/ app users) and assets (web server, app server)
  - Maps out intended targets and players

Traditional Vector Based Attack Types

- **Buffer overflows attacks (aka string injection)**
- **Code injection attacks**: also known as "code poisoning attacks“ examples:
  - Cookie poisoning attacks
  - HTML injection attacks
  - File injection attacks
  - Server pages injection attacks (e.g. ASP, PHP)
  - Script injection (e.g. cross-site scripting) attacks
  - Shell injection attacks
  - SQL injection attacks
  - XML poisoning attacks

From: ITtoolbox Wiki http://it.toolbox.com/wiki/index.php/Attack_vector

Attack Vectors

Attacks to by-pass filtering, exploit IV vulnerability patterns:

1. Input=>Output==XSS
2. Input=>Query (SQL, LDAP) ==(SQL, LDAP) injection
3. Input=>Code== Code injection
4. Input=>XML doc == XML injection
5. Input=>OS command==OS command injection
6. Input=> Fixed buffer or format string== overflow

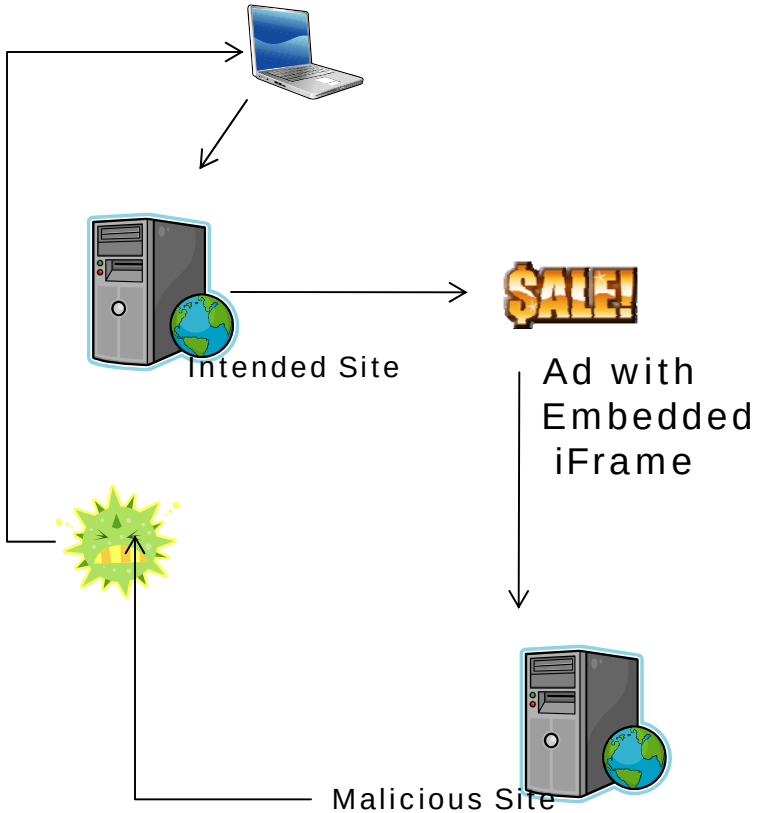
New Web 2.0 Attack Vectors

- Cross-site scripting in AJAX
- XML Poisoning
- Malicious AJAX code execution
- RSS Atom Injection
- WSDL scanning and enumeration
- Client validation in AJAX routines
- Web service routing issues
- Parameter manipulation with SOAP
- XPATH injection in SOAP message
- RIA thick client binary vector

FromTop 10 Web 2.0 Attack Vectors http://www.net-security.org/article.php?id=949&p=4

Need to include Cybercrime attack vectors

- IFRAME injection (InLine Frame Injection)
  - Trusted sites with malicious banner ads
  - Browser vulnerabilities in handling iFrame tags
  - Leverages blackhat SEOs in order to drive traffic to vulnerable sites
  - Growing attack vectors for malware propagation
- SQL injection still a key factor
  - Continues to be successful attack vector
  - Exploits poor validation, type-safe values, parameterized queries
  - Poor web programming fosters problem
    - SELECT *fieldlist* FROM *table* WHERE *field* = '$EMAIL';
  - Simple mistakes become costly

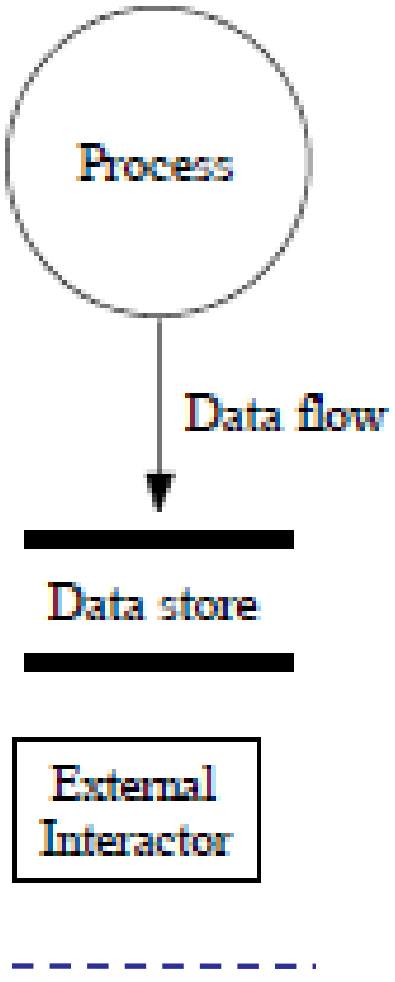


Architecture analysis via threat modeling

DFD for Threat Models

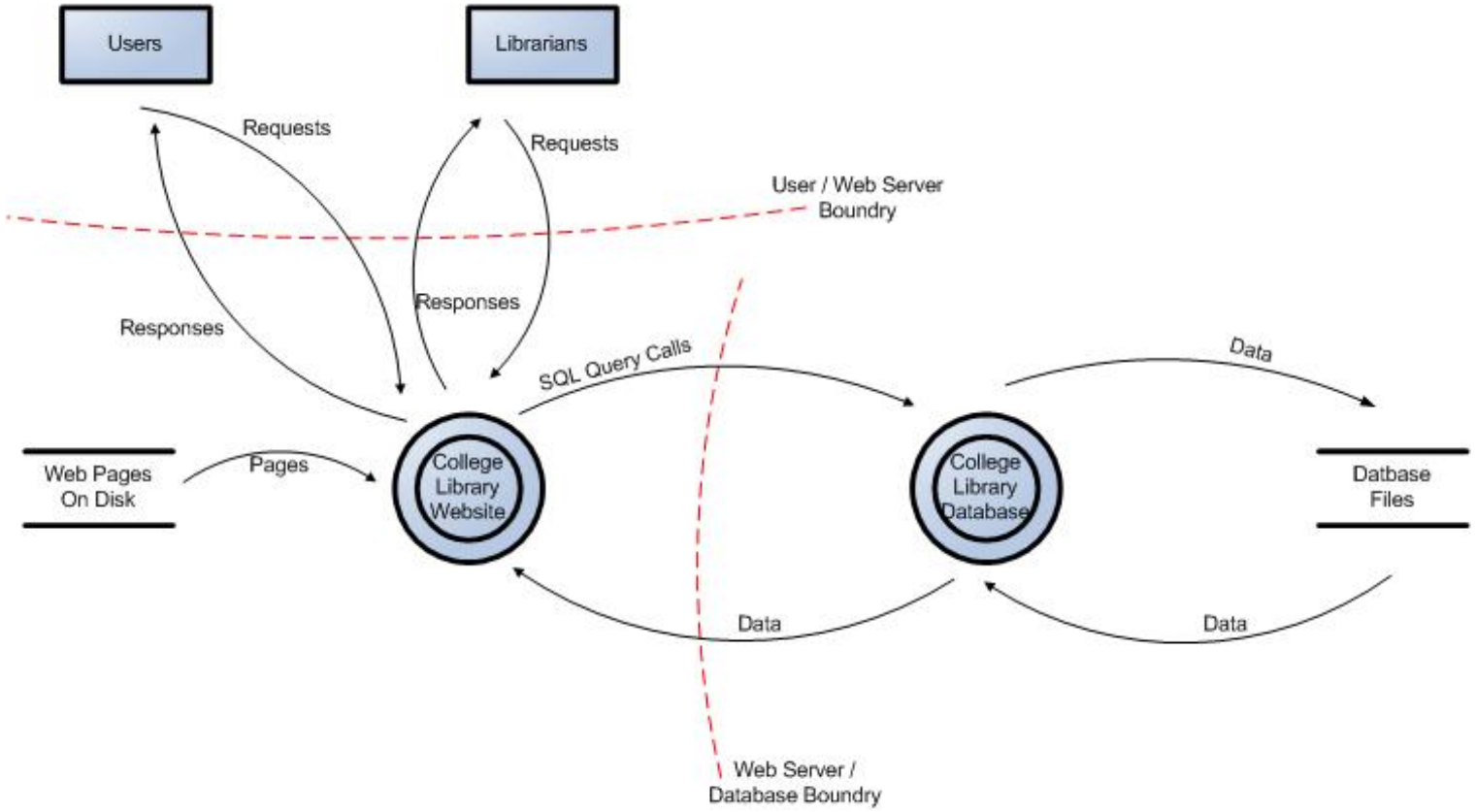
- Describe how the data interacts with the system at different layers/tiers by diagramming the following elements:
  - External entities (users, web browser, email-client, external site/system)
  - Processes (Servers: Web, Application, DB server, functions: input validation, authentication, authorization, session management, encryption)
  - Data flows (inbound/outbound function calls, call types, type of data and protocol)
  - Data stores (file, database, any data at rest)
  - Trust levels and trust boundaries

DFD Symbols Used For Threat Models

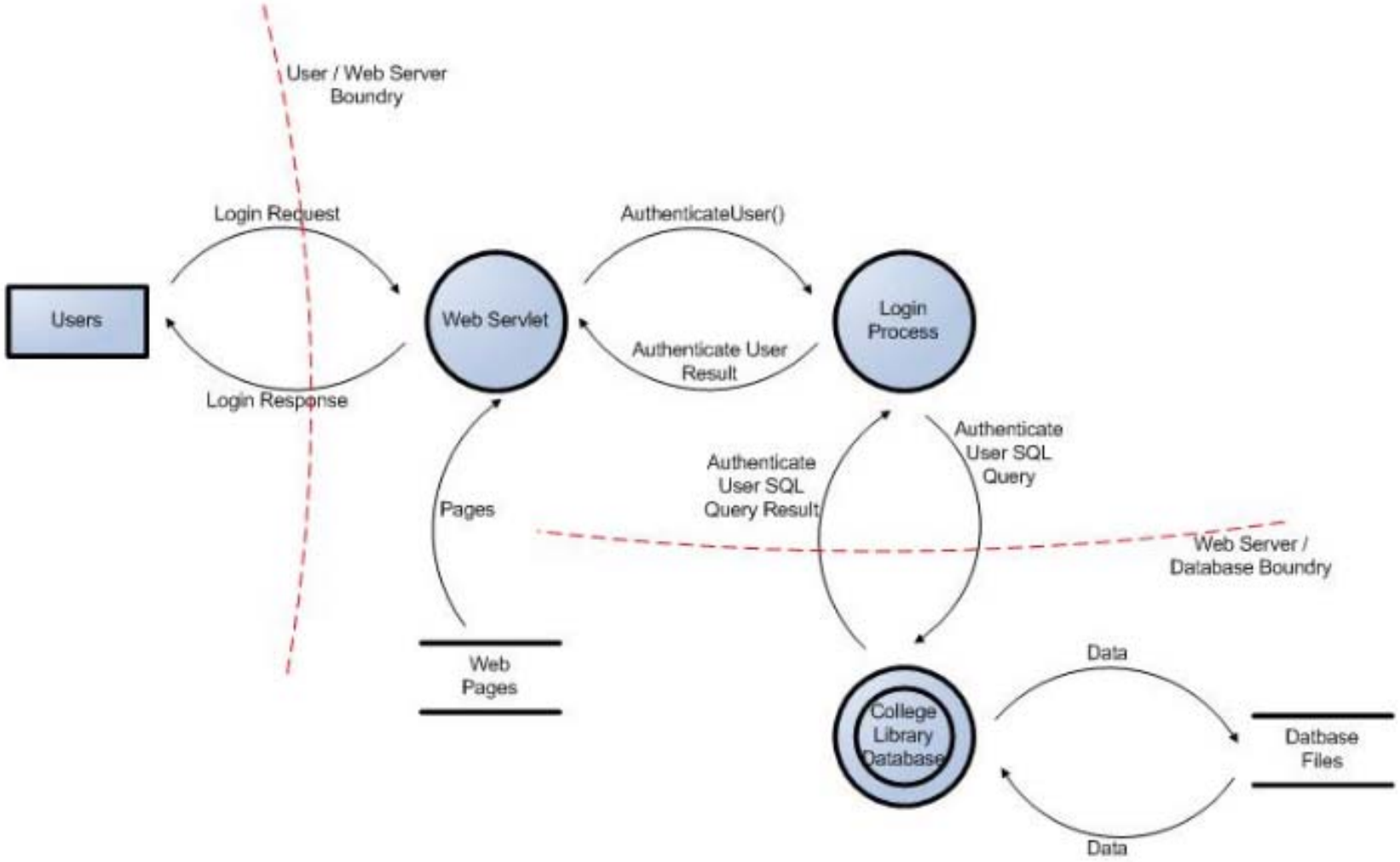


- A process: represents a task in the system that processes data
- A data flow represents a path for data from between elements
- A data store represents a repository where data is saved or retrieved, but not changed.
- An external entity represents a source/sink for the data
- A privilege boundary represents a change of privilege as the data flows through the application

College Library Web Site: DFD Level 2 Visualizing End to End Data Flows



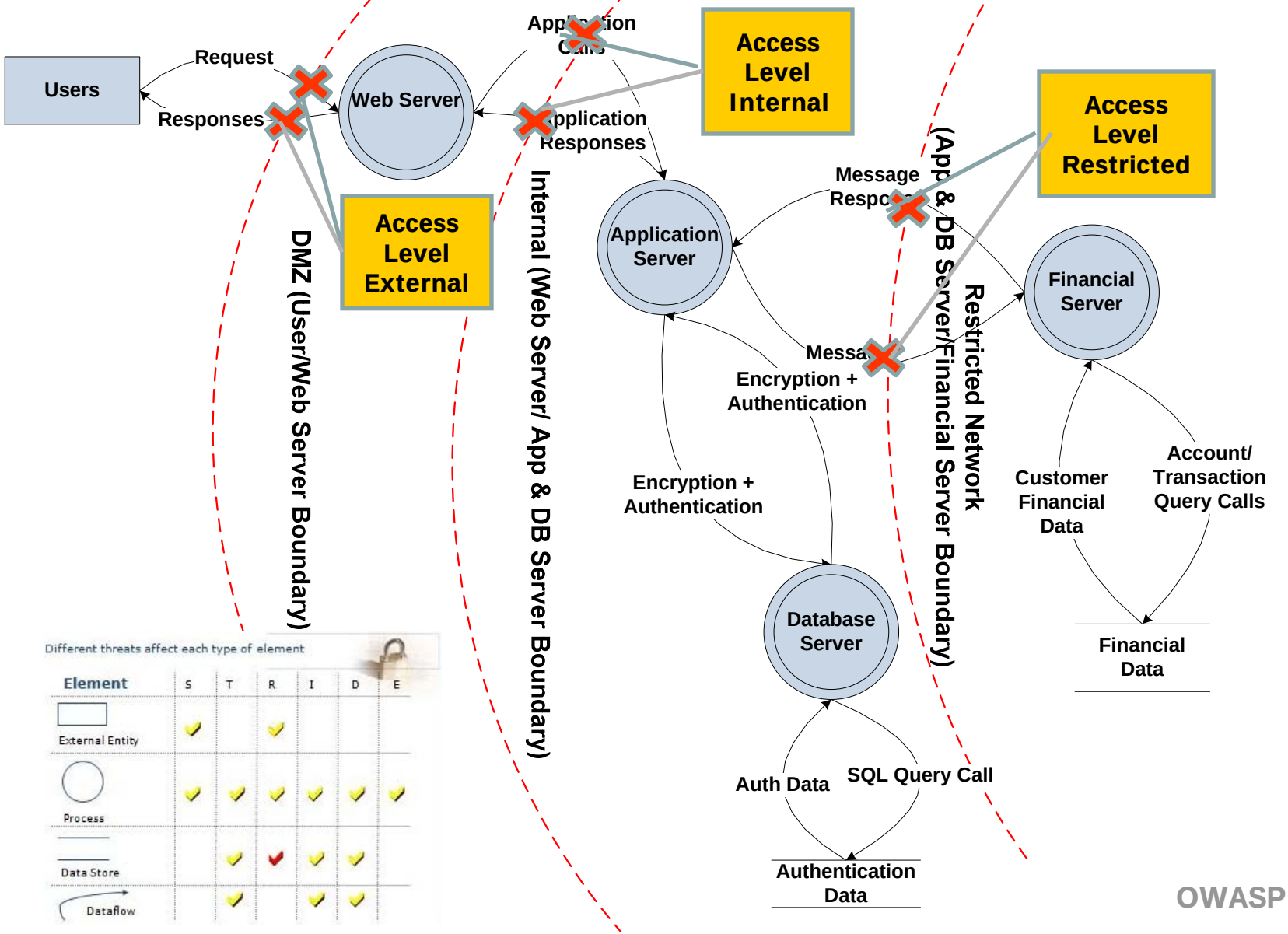
College Library Web Site: DFD Level 1 Visualizing Login Process



DFD Threat Analysis Objective

- Identify entry and the exit points and the access levels
- Enumerate the threats to the application elements
- Identify the impact for the most likely attack vectors
- Identify mitigations to the previously identified attack vectors by locating the countermeasures

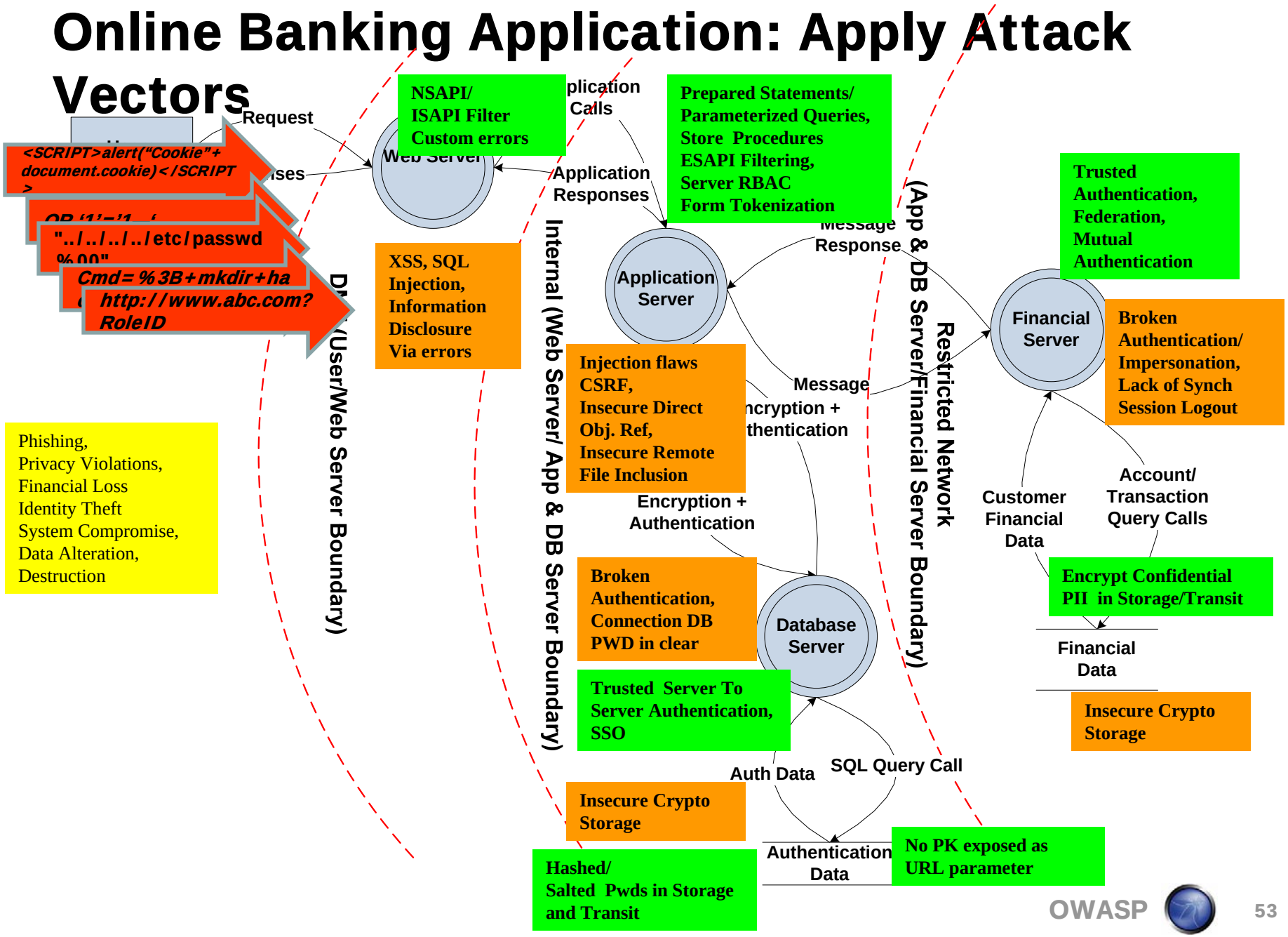
On-line Banking Application: Threat Analysis



Threats and Mitigation Techniques

STRIDE Threat & Mitigation Techniques List	
Threat Type	**Mitigation Techniques**
Spoofing Identity	1. Appropriate authentication 2. Protect secret data 3. Don't store secrets
Tampering with data	1. Appropriate authorization 2. Hashes 3. MACs 4. Digital signatures 5. Tamper resistant protocols
Repudiation	1. Digital signatures 2. Timestamps 3. Audit trails
Information Disclosure	1. Authorization 2. Privacy-enhanced protocols 3. Encryption 4. Protect secrets 5. Don't store secrets
Denial of Service	1. Appropriate authentication 2. Appropriate authorization 3. Filtering 4. Throttling 5. Quality of service
Elevation of privilege	1. Run with least privilege

Online Banking Application: Apply Attack Vectors


Request
<SCRIPT>alert("Cookie"+ document.cookie)</SCRIPT >
"../../../etc/passwd %00"
Cmd=%3B+mkdir+ha
http://www.abc.com?RoleID
NSAPI/ ISAPI Filter Custom errors
Application Calls
Application Responses
Prepared Statements/ Parameterized Queries, Store Procedures ESAPI Filtering, Server RBAC Form Tokenization
XSS, SQL Injection, Information Disclosure Via errors
Application Server
Message Response
Financial Server
Trusted Authentication, Federation, Mutual Authentication
Broken Authentication/ Impersonation, Lack of Synch Session Logout
Injection flaws CSRF, Insecure Direct Obj. Ref, Insecure Remote File Inclusion
Message
Encryption + Authentication
Encryption + Authentication
Customer Financial Data
Account/ Transaction Query Calls
Encrypt Confidential PII in Storage/Transit
Financial Data
Insecure Crypto Storage
Broken Authentication, Connection DB PWD in clear
Database Server
Trusted Server To Server Authentication, SSO
Auth Data
SQL Query Call
Insecure Crypto Storage
Authentication Data
No PK exposed as URL parameter
Hashed/ Salted Pwds in Storage and Transit
Phishing, Privacy Violations, Financial Loss Identity Theft System Compromise, Data Alteration, Destruction
(User/Web Server Boundary)
Internal (Web Server/ App & DB Server Boundary)
Restricted Network
(App & DB Server/Financial Server Boundary)

Countermeasure Identification

ASF Threat & Countermeasures List	
Threat Type	**Countermeasure**
Authentication	1. Credentials and authentication tokens are protected with encryption in storage and transit 2. Protocols are resistant to brute force, dictionary attack and replay attacks 3. Strong password policies are enforced 4. Trusted server authentication is used instead of SQL authentication 5. Passwords are stored with salted hashes 6. Password resets do not reveal password hints and valid usernames 7. Account lockouts do not result in a denial of service attack
Authorization	1. Strong ACLs are used for enforcing authorized access to resources 2. Role-based access controls are used to restrict access to specific operations. 3. The system follows the principle of least privilege for user and service accounts 4. Privilege separation is correctly configured within the presentation, business and data access layers.
Configuration Management	1. Least privileged processes are used and service accounts with no administration capability 2. Auditing and logging of all administration activities is enabled 3. Access to configuration files and administrator interfaces is restricted to administrators
Data Protection in Storage and Transit	1. Standard encryption algorithms and correct key sizes are being used 2. Hashed message authentication codes (HMACs) are used to protect data integrity 3. Secrets (e.g. keys, confidential data) are cryptographically protected both in transport and in storage 4. Built-in secure storage is used for protecting keys 5. No credentials and sensitive data are sent in clear text over the wire

https://www.owasp.org/index.php/Application_Threat_Modeling

Mitigation strategies against cybercrime attacks

Mitigations Against Cyber threats

- Unify security efforts as part of threat modeling
  - Risk Assessments – Info from application risk assessments can refine future threat models
  - Vuln Assessments – help fuel vuln database for each application or application component
  - Static Analysis – provides additional vuln information to classify poor coding practices as vulns that could be exploited
  - Pen Testing – provides probability values that helps define where countermeasures are most needed
  - Security Architecture – opportunity to define security requirements for platforms, softwares
- Mitigation principles
  - Adopt cyber-intelligence and cyber threat analysis
  - Adopt Application Threat Modeling
  - Promote secure architecture design
    - Document strong architecture principles (eventually should be part of compliance requirements

Secure By Design

- Securing The Network
  - Access Controls: Default Deny vs. Default Permit
  - Auditing and Logging (A&L)
  - IDS/IPS, Firewall/ACLs
  - Patching
- Securing The Web server
  - Hardening and Locking
  - Secure Configuration Management
  - Auditing and Logging
- Securing The Application Server
  - Mutual SSL Authentication, WS-Security, Secure XML, Secure Session Management, Auditing & Logging
- Securing The Database
  - Hardening, Extended Store Procedures, Access Privileges,

S-ODBC, Data Protection in Storage: Hashing/Encryption, A&L

General Security Design Principles

1. Implement Authentication With Adequate Strength
2. Enforce Least Privilege
3. Protect Sensitive Data In Storage, Transit And Display
4. Enforce Minimal Trust
5. Trace and Log User Actions And Security Events
6. Fail Securely And Gracefully
7. Apply Defense in Depth
8. Apply Security By Default
9. Design For Simplicity, KISS Principle
10. Secure By Design, Development and Deployment
11. Secure As The Weakest Link
12. Avoid Security By Obscurity

Threat Modeling and Risk Decision Making

- Threats can be resolved by:
  - Risk Acceptance - doing nothing
  - Risk Transference - pass risk to an externality
  - Risk Avoidance - removing the feature/component that causes the risk
  - Risk Mitigation - decrease the risk
- Mitigation strategies should be examined for each threat
- Mitigations should be chosen according to the appropriate technology
- Resolution should be decided according to risk level and cost of mitigations

Risk Mitigation Strategy Best Practices

1. Translate the identified technical risk issues into business risk issues
2. Determine whether standards compliance is violated
3. Determine whether other compensating security controls and mitigation factors are present
4. Determine which protective measure (e.g. security control, policy measures) should be in place to mitigate the threat, how should be implemented and when
5. Determine short term and long term solutions

Thank You!

Further References